AMENDMENTS TO CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A high speed search method in a speech encoder using an order character of LSP (Line Spectrum Pair) parameters in an LSP parameter quantizer using SVQ (Split Vector Quantization) used in a low-speed transmission speech encoder, the high-speed search method comprising the steps of:

rearranging a first codebook by replacing the first codebook with a new codebook in which a number of code vectors in the new codebook are arranged in an order according to an element value of a reference row of the first codebook for determining a range of code vectors to be searched; and

determining a search range by using an order character between a given target vector and an arranged code vector to obtain an optimal code vector,

wherein the rearranging step comprises the steps of:

selecting the reference row in the first codebook by using a plurality of voice data, and then determining an optimal arrangement position (Nm) in which an average search range is minimized; and

replacing the first codebook with the new codebook in which a number (Lm) of code vectors in the new codebook are arranged in a descending order according to the element value of a selected said reference row.

2. (Canceled)

3. (Currently Amended) ~~The~~ A high-speed search method ~~as claimed in claim 1~~ in a speech encoder using an order character of LSP (Line Spectrum Pair) parameters in an LSP parameter

quantizer using SVQ (Split Vector Quantization) used in a low-speed transmission speech encoder, the high-speed search method comprising the steps of:

rearranging a first codebook by replacing the first codebook with a new codebook in which a number of code vectors in the new codebook are arranged in an order according to an element value of a reference row of the first codebook for determining a range of code vectors to be searched; and

determining a search range by using an order character between a given target vector and an arranged code vector to obtain an optimal code vector,

wherein ~~the code vector-obtaining step~~ obtaining an optimal code vector comprises the ~~step~~ steps of:

determining the search range by forward and backward comparison of the element value of the reference row in the first codebook and element values of positions before and after a reference position in the target vector; and

obtaining an error criterion ($E_{l,m}$) having high computational complexity by using the following equation only within the determined search range:

$$\mathbf{E}_{l,m} = (\mathbf{p}_m - \mathbf{p}\}_{l,m}\}^T \mathbf{W}_m(\mathbf{p}_m - \mathbf{p}\}_{l,m})$$

$$0 \le m \le M - 1$$

$$1 \le l \le L_m$$

where **p** is an LSP code vector divided into M sub-vectors, each of which consists of L_m code vectors,

where $\mathbf{p}_m$ is a target vector to search the m^{th} codebook, and $\mathbf{p}\}_{l,m}$ corresponds to an l^{th} code vector in a codebook for an m^{th} sub-vector,

where l,m in the subscript of $E_{l,m}$ are indices that represent the *l*th index of the *m*th codebook, *i.e.*, the letters "*l*" and "m," ~~and~~

where superscript T designates the transpose of $(\mathbf{p}_m - \mathbf{p}\}_{l,m}\}$ for purposes of determining the dot product of $(\mathbf{p}_m - \mathbf{p}\}_{l,m}\}$ and $\mathbf{W}_m (\mathbf{p}_m - \mathbf{p}\}_{l,m}\}$ in order to calculate the least-mean-square error $E_{l,m}$, and

where $\mathbf{W}_m$ is a weighting matrix for the m^{th} sub-vector and obtained by a non-quantized LSP code vector **p**.

4. (Currently Amended) The high-speed search method as claimed in claim 3,

wherein the search range is an average number with which an element value of the n^{th} row in the first codebook and element values in the $n+1^{th}$ and $n-1^{th}$ positions of the target vector satisfy the order character.

5. (Currently Amended) A high-speed search method in the G.729 fixed codebook with decreased computational complexity without loss of tone quality, the high-speed search method comprising the steps of:

arranging position indexes of tracks (t_0, t_1, t_2) in a descending order according to a correlation level (d'(n));

determining a range to search a track (t3) according to the indexes arranged in a descending order; and

canceling ~~the~~ detecting and searching processes for indexes which have a low probability.

6. (Currently Amended) The high-speed search method in the G.729 fixed codebook as claimed in claim 5, wherein the arranging step comprises the step of:

comparing correlation vectors of all of pulse position indexes in each track to arrange the position indexes in a descending order.

7. (Currently Amended) The high-speed search method in the G.729 fixed codebook as claimed in claim 5, wherein the search range-determining step comprises the steps of:

adding correlation values of each pulse position index for a pulse position index combination of the tracks (t_0, t_1, t_2); and

comparing an added result with a threshold (Cth) determined before searching the fixed codebook to search track (t3) using ~~an~~ the added result more than the threshold.

8. (Currently Amended) The high-speed search method in the G.729 fixed codebook as claimed in claim 5, wherein the canceling step comprises the step of:

canceling the searching ~~process~~ processes for the range where an added result is less than a threshold.